

## [IR2018-19-HW1] 1183732 Davide Martini

### 1. Descrizione del lavoro svolto

Lo sviluppo dell'homework è stato effettuato in ambiente *Linux, Ubuntu 18.04*, tramite l'utilizzo di:

- *terrier-core-4.4* per l'indicizzazione dei documenti e la generazione delle varie run [2].
- *trec\_eval 9.0.4* per l'ottenimento delle misure di valutazione dei vari sistemi [3].
- *os* libreria di *Python* per le chiamate di sistema [4].
- *matplotlib* libreria in *Python* per la creazione dei grafici riassuntivi [5].
- *statsmodel* libreria di *Python* per il calcolo del test ANOVA 1-way e il Tukey HSD test [6].
- *numpy, scipy* librerie *Python* accessorie [7].
- *PyCharm 2018.3* programma utilizzato per lo sviluppo ed il test dello script [8].

È stato inoltre prodotto uno script in *Python 3.7* per automatizzare l'esecuzione dell'indicizzazione, la produzione delle varie run, il calcolo delle misure, produrre il test statistico ANOVA 1-way, il Tukey HSD test e per ottenere i plot finali. Lo script è contenuto nella repository con il nome di '*ir\_script.py*'. Di seguito verrà riportata la struttura del codice con una breve spiegazione del suo contenuto.

Per automatizzare l'esecuzione di tutto il workflow necessario per ottenere i risultati, il codice esegue i seguenti passi:

- I. Viene impostata una cartella principale dove è contenuta la collezione di documenti, gli eseguibili dei software *terrier* e *trec\_eval* e dove verranno salvati i risultati ottenuti.
- II. Viene inizializzato il software *terrier* e vengono modificate le proprietà per l'indicizzazione e per l'esecuzione delle run. Questo passaggio viene ripetuto per tutti i modelli da analizzare. Sono stati salvati i vari file degli indici nella cartella '*indexes/run*' e i file delle run nella cartella '*indexes/run/eval*'.
- III. Viene eseguita la valutazione tramite il software *trec\_eval* sui file risultanti delle run. I file contenenti le misure per la valutazione sono stati successivamente salvati nella cartella '*indexes/run/eval*'.
- IV. È stata creata una struttura per contenere le varie misure di valutazione ottenute dal software *trec\_eval*, in modo da avere un facile reperimento di queste per i test statistici e per i successivi plot.
- V. Sono stati creati dei file, per ogni misura, in cui ogni colonna rappresenta una run.
- VI. È stata successivamente effettuata l'ANOVA 1-way ed il Tukey HSD test.
- VII. Sono stati infine prodotti i plot per ogni topic e run delle misure *P(10)* e *Rprec*. Un grafico contenente il valore della *MAP* per ogni modello analizzato.

### 2. Risultati ottenuti

In *Tabella 1* vengono riportati il numero di documenti indicizzati e il numero di parole presenti nel vocabolario, in modo da notare come vari l'utilizzo o meno delle stopword e del Porter Stemmer.

I vari sistemi con le diverse impostazioni, sono riportati con una sigla:

- *BM25*: modello BM25 con stopword e Porter Stemmer.
- *BM25\_stem*: modello BM25 senza stopword, con Porter Stemmer.
- *TF\_IDF*: modello TF\_IDF con stopword e Porter Stemmer.
- *TF\_IDF\_not*: modello BM25 senza stopword e Porter Stemmer.

	TF_IDF	BM25	BM25_stem	TF_IDF_not
documenti indicizzati	528155	528155	528155	528155
dimensione del dizionario	738439	738439	738643	840517

**Tabella 1.** Statistiche dei vari indici

È stato sviluppato il test statistico ANOVA 1-way, Tukey HSD pairwise test e Tukey HSD multiple comparisons.

L'obiettivo del test statistico ANOVA 1-way è capire se i vari sistemi di reperimento analizzati avessero, oppure no, la stessa media. Nello sviluppo del test, il valore di soglia  $\alpha = 0.05$  permette di rifiutare o meno l'ipotesi che i 4 sistemi abbiano la stessa media. I valori ottenuti sono presenti nel file '*run/plot/anova.txt*' e vengono riportati in *Tabella 2*.

$F_{\text{stat}}$	$\sim 0.0996$
$p = P[F \geq F_{\text{stat}}   H_0]$	$\sim 0.9601$

**Tabella 2.** ANOVA 1-way.

Si nota quindi che l'ipotesi non viene rifiutata e quindi i 4 sistemi di reperimento dell'informazione hanno la stessa media.

Per quanto riguarda il Tukey HSD pairwise test, i risultati ottenuti sono riportati nella *Tabella 3*.

Multiple Comparison of Means - Tukey HSD, FWER=0.05					
group 1	group 2	meandiff	lower	upper	reject
BM25	BM25_stem	0.0029	-0.0816	0.0874	False
BM25	TF_IDF_not	-0.0135	-0.0980	0.0710	False
BM25	TF_IDF	-0.0006	-0.0851	0.0838	False
BM25_stem	TF_IDF_not	-0.0164	-0.1008	0.0681	False
BM25_stem	TF_IDF	-0.0035	-0.0880	0.0810	False
TF_IDF_not	TF_IDF	0.0128	-0.0716	0.0973	False

**Tabella 3.** Tukey HSD pairwise test.

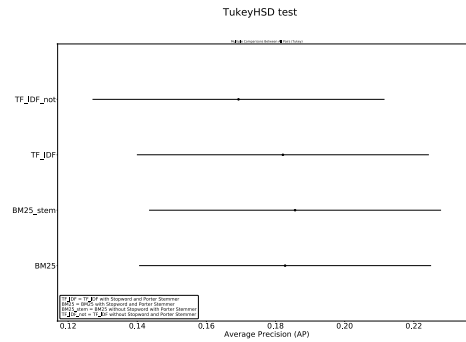
Dal Tukey HSD test si nota che ogni coppia di sistemi è simile e c'è sempre intersezione negli intervalli di confidenza. Per avere una visione d'insieme è stato effettuato un confronto multiplo tra i vari modelli riportato in *Figura 1* e nel file '*run/plot/TukeyHSDtest.svg*'.

I sistemi adottati quindi sono molto simili, anche se l'indicizzazione del metodo *TF\_IDF\_not* è risultata più lunga in quanto sono state tenute in considerazione tutte le parole e queste non sono state sottoposte al processo di stemming.

Per quanto riguarda l'analisi della misura *Rprec* il confronto dei grafici, presenti nella cartella '*run/plot*' con nome '*RprecSYSTEM.svg*', dove *SYSTEM* indica il sistema considerato, indica una lieve differenza tra i metodi *BM25*, *BM25\_stem*, *TF\_IDF*, ma evidenzia un forte calo in *TF\_IDF\_not*. Questa differenza non è presente in tutti i topic ma è abbastanza accentuata nei topic dove si verifica.

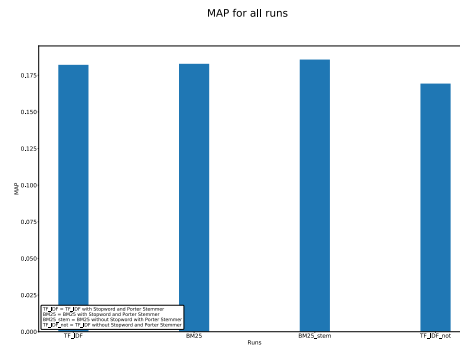
I valori ottenuti per la misura *P(10)*, presenti nei grafici nella cartella '*run/plot*' con nome '*P\_10SYSTEM.svg*' dove *SYSTEM* indica il sistema considerato, mostrano un andamento abbastanza simile per i metodi che adottano *BM25*, *BM25\_stem*, *TF\_IDF*, ma *TF\_IDF* ha un andamento simile per alcuni topic e un valore minore della *Precision* per gli altri.

Da queste due misure, anche se il test di Tukey non ha mostrato grandi differenze, si può definire  $TF\_IDF\_not$ , come il peggiore dei 4 sistemi per quanto riguarda l'aspetto di efficacia nel reperimento dei documenti.



**Figura 1.** Tukey HSD test - Multiple Comparisons.

In *Figura 2*, viene riportato il valore della  $MAP$  per ogni metodo. Notiamo che  $TF\_IDF\_not$  ha una  $MAP$  più bassa tra tutti mentre, come nelle considerazioni precedenti, gli altri 3 assumono valori molto simili.



**Figura 2.** MAP.

Durante la fase di indicizzazione è stato conteggiato il tempo necessario per la sua esecuzione per ogni sistema, i quali vengono riportati in *Tabella 4*. I numeri rappresentano i secondi utilizzati per effettuare l'indicizzazione dei documenti.

TF_IDF	BM25	BM25_stem	TF_IDF_not
260.3225	260.3225	293.6788	294.5869

**Tabella 3.** Tempo di indicizzazione della collezione (sec).

Si può notare che ovviamente nei due sistemi nei quali non vengono eliminate le stopwords il tempo di esecuzione è maggiore. Questo è già un elemento che potrebbe incidere nella scelta di un metodo rispetto ad un altro.

### 3. Conclusioni

Come riportato nel test ANOVA1-way la media dei vari sistemi risulta essere la stessa, ed il Tukey HSD pairwise test e il multiple comparisons, mettono in evidenza che non ci sono delle differenze sostanziali.

Occorre però notare che i valori di  $Rprec$  e  $P(10)$  sono molto più bassi in  $TF\_IDF\_not$ , il quale ha inoltre un dizionario più ampio ed un tempo di indicizzazione maggiore rispetto agli altri per il fatto che vengono considerate tutte le parole presenti nei documenti e non viene applicata la fase di stemming. Questo risulta quindi, il sistema peggiore da utilizzare tra i 4 analizzati.

Per quanto riguarda i 3 restanti, un tempo di esecuzione di circa il 10% in più rispetto ai due metodi che eliminano le stopwords è un fattore negativo per  $BM25\_stem$ , che applicato su una collezione di dimensioni maggiori verrebbe scartato in quanto l'efficienza di un IRS è un fattore molto importante da considerare.

Tra  $TF\_IDF$  e  $BM25$ , nei quali vengono eliminate le stopwords e viene applicato il Porter Stemmer, il tempo di indicizzazione è lo stesso e i risultati per quanti riguarda  $MAP$ ,  $Rprec$  e  $P(10)$  sono molto simili. Quindi tra i 4 sistemi analizzati questi si sono rilevati quelli con una migliore efficienza ed efficacia.

### 4. Riferimenti

- [1] Github: <https://github.com/davidemartini/Information-Retrieval>
- [2] Terrier: <http://terrier.org/>
- [3] Trec\_eval: [https://github.com/usnistgov/trec\\_eval](https://github.com/usnistgov/trec_eval)
- [4] os: <https://docs.python.org/3/library/os.html?highlight=os#module-os>
- [5] matplotlib: <https://matplotlib.org/>
- [6] statsmodel: <https://www.statsmodels.org/stable/index.html>
- [7] numpy e scipy: <https://docs.scipy.org/doc/>
- [8] PyCharm: <https://www.jetbrains.com/pycharm/>